Northeastern University - Seattle

Khoury College of Computer Sciences

CS6200 Information Retrieval Fall 2019

Lecture 7: Query Refinement

Oct 17, 2019

Administrivia

- Assignment 2
  - Some of you are happily chugging away on this, great!
  - Hope you all have started on it.
    Do NOT wait till the last minute.
  - Due 9am Pacific, Oct 21st
  - Note: Use the assignment as a check list, ensure you submit the right files, with the specified names.
  - Questions?

Quiz 5 Discussion .. 1

1. Query arithmetic fun:

The following queries were tried in Bing:

Sherlock Holmes => 1,960,000 results

Sherlock Holmes Detective => 7,600,000 results

(Results go up as Detective is a relatively common term)

sherlock Holmes detective +"Baskervilles" => 174,000 results

(Lesser results as Baskervilles is a small subset of the related corpus)

sherlock Holmes detective +"Baskervilles“ –“Watson” => 107,000 results

(Slightly lesser results as Sherlock Holmes is incomplete without Watson)

sherlock Holmes site:pcgamer.com => 222 results

(Very specific site causes drastically confined results)

Quiz 5 Discussion .. 2

2. Results caching may not be useful for:

    Stock market prices, flight schedules, traffic directions

3. Lambda = 0 → no random choice, just follow page links, may miss disconnected pages (with no inlinks), may get stuck on pages with bad or no outlinks.

Lambda = 0.5 → Random page selected with some frequency, but popular pages still preferred with same frequency

Lambda = 1 → (almost) fully random choice, no importance given to links on page, popularity not as important

Read the question, please. At least one person answered for lambda = 0.15, 0.5 and 1, another for 1, 0, 1.

Quiz 5 Discussion .. 3

4. Timeouts are required to ensure we do not get stuck waiting for some server to respond etc.

5. Advanced operators

Movie:, stocks:, allintitle:, intitle:, OR, |, allinurl:,

allintext: rules of civility, inanchor:, 1…1000 inspired,

Filetype:, cache, language:, contains:, location:, map:,

Define:

Quiz 6

- Released today, Oct 17
- Due 9am Pacific, Oct 24
  - Note non-standard due date

Overview

Queries & Information Needs

Query Transformations/Refinement

- Query-based Stemming
- Spell Checking & Spelling Suggestions
- Query Expansion
- Relevance Feedback & Pseudo-Relevance Feedback
- Context & Personalization

QUERIES & INFORMATION NEEDS

Bing Google

Information Needs

- An *information need* is the underlying cause of the query that a person submits to a search engine
  - sometimes called *information problem,* because information need is generally related to a task
- Categorized on a variety of dimensions, e.g.
  - number of relevant documents sought
  - type of information required
  - type of task that led to need for information

Queries and Information Needs ..1

A query can represent very different information needs

- e.g. Las Vegas on Expedia, Kabul on Yelp!
- May require different search techniques and ranking algorithms for best rankings

Queries and Information Needs ..2

A query may be a poor representation of information need

- User may find it difficult to express the information need
- In part because users encouraged to enter short queries
  - by the search engine interface [Textbox: SPL 26 chars, Bing ~70 chars, Google ~52 chars]
  - the fact that longer queries often don't work as well

Interaction

- Interaction with the system
  - during query formulation and reformulation
  - while browsing the result
- Key aspect of *effective retrieval*
  - users cannot change *ranking algorithm* but can change results through ***interaction***
  - can refine description of information need
    - e.g., same initial query, different information needs
    - how does user describe what they don't know?

ASK Hypothesis

Nick Belkin

- Belkin et al (1982) proposed a model called Anomalous State of Knowledge
- ASK hypothesis:
  - Sometimes difficult for people to define exactly what their information need is, because that information is a gap in their knowledge
  - Search engine should look for information that fills those gaps
- Interesting ideas, but Catch 22 – little practical impact (yet)

https://comminfo.rutgers.edu/directories/all-sci/B

Keyword Queries

- Query languages in the past were designed for professional searchers (*intermediaries*)

User query:
Are there any cases which discuss negligent maintenance or failure to maintain aids to navigation such as lights, buoys, or channel markers?

Intermediary query:
NEGLECT! FAIL! NEGLIG! /5 MAINT! REPAIR! /P NAVIGAT! /5 AID EQUIP! LIGHT BUOY "CHANNEL MARKER"

a or b or c within 5 words of ...

in the same para

phrase

Keyword Queries

Simple, *natural language* queries were designed to enable everyone to search

Current search engines do not perform well (in general) with natural language queries

People ‘trained’ (in effect) to use keywords

compare average of about 2.3-2.7 words/web query to average of 30 words/CQA query

Keyword selection not always easy

query refinement techniques can help

Evolution of Search

- Search has moved
  from relying on search intermediaries
  to putting more ‘intelligence’ into search,
  or enabling users to affect search

- Techniques to be discussed relate to that
  - transforming queries [today]
  - changing how results are displayed [Lecture #10]

QUERY-BASED STEMMING

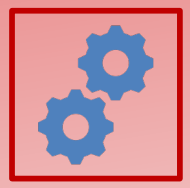
Query-Based Stemming

- Make decision about stemming at query time rather than during indexing
  - improved flexibility, effectiveness
- Query is expanded using word variants
  - documents are not stemmed
  - e.g., [rock climbing]: expanded with [climb], not stemmed to “climb”

  [rock (climbing OR climb OR climbed…)]

  But how do we know which words to use in expanded query?

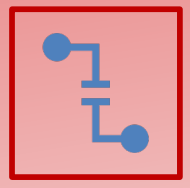
Stem Classes



A *stem class* is the group of words that will be transformed into the same stem by the stemming algorithm

generated by running stemmer on large corpus

e.g., Porter stemmer on TREC News



Examples:

/**bank** banked banking bankings banks

/**polic** polical polically police policeable policed policement policer
policers polices policial policicallu policier policiers policies policing
policization policize policly policy policying policys

Stem Classes

- Stem classes are often too big and inaccurate
  - e.g. policy and police → polic ?
- Modify using analysis of *word co-occurrence*
- *Assumption:*
  - Word variants that could substitute for each other should co-occur often in documents

Modifying Stem Classes

1. For all pairs of words in stem classes, count how often they co-occur in text windows W (50-100 words) or documents.
2. Compute co-occurrence/association metric for each pair (= how strong association is between words)
3. Construct graph: vertices represent words, edges are between words with metric > T
4. Find connected components in graph – these are new stem classes

Modifying Stem Classes

- Dice's Coefficient: a term association measure:

$$2 * n_{ab} / (n_a + n_b)$$

where n_x is the number of windows containing x

- Two vertices are in the same connected component of a graph if there is a path between them
  - forms word *clusters*
- Sample output of modification:
  - /policies policy
  - /police policed policing
  - /bank banking banks

SPELL CHECKING

Spell Checking .. 1

- Important part of query processing
  - 10-15% of all web queries have spelling errors
- Errors include typical word processing errors but also many other types, e.g.

poiner sisters
brimingham news
catamarn sailing
hair extenssions
marshmellow world
miniture golf courses
psyhics
home doceration

realstateisting.bc.com
akia 1080i manunal
ultimatwarcade
mainscourcebank
dellottitouche

Spell Checking .. 2

- Different from document spell-check:
  e.g. “For each token, find set of alternative words based on certaint ypes of erros, using dicoitonary based on bvoth query log and trusted dictionar.”

- Large number of names (people, places, organizations) -- not in any dictionary
  - Junglee, Flickr, Precious Ramotswe, Thiruvananthapuram, Chomolungma

Spell Checking .. 3

Basic approach: suggest corrections for words not found in *spelling dictionary*

but issue when a misspelling is really another valid word e.g. "the Chase Ban teller" ; ban → bank?

Suggestions found by comparing word to words in dictionary using similarity measure

Most common similarity measure is *edit distance*

number of operations required to transform one word into the other

Edit Distance .. 1

Damerau-*Levenshtein* distance

- counts the minimum number of insertions, deletions, substitutions, or transpositions of single characters required
- e.g., Damerau-Levenshtein distance 1

  extenssions $\rightarrow$ extensions(insertion error)
  poiner $\rightarrow$ pointer (deletion error)
  marshmellow $\rightarrow$ marshmallow $(\text{substitution error})$
  brimingham $\rightarrow$ birmingham $(\text{transposition error})$

- distance 2

  doceration $\rightarrow$ deceration
  deceration $\rightarrow$ decoration

Edit Distance .. 2

Many ways to speed up calculation of edit distances

- restrict to words starting with same character
- restrict to words of same or similar length
- restrict to words that sound the same

Last option uses a *phonetic code* to group words

- e.g. Soundex

Soundex Code

1. Keep the first letter (in upper case).
2. Replace these letters with hyphens: a,e,i,o,u,y,h,w.
3. Replace the other letters by numbers as follows:
   - 1: b,f,p,v
   - 2: c,g,j,k,q,s,x,z
   - 3: d,t
   - 4: l
   - 5: m,n
   - 6: r
4. Delete adjacent repeats of a number.
5. Delete the hyphens.
6. Keep the first three numbers or pad out with zeros.

extenssions → E235; extensions → E235
marshmellow → M625; marshmallow → M625
brimingham → B655; birmingham → B655
poiner → P560; pointer → P536

Spelling Correction Issues

- Ranking the corrections
  - "Did you mean..." feature requires accurate ranking of possible corrections
- Context
  - Choosing right suggestion depends on context (other words)
  - e.g., *lawers* → *lowers, lawyers, layers, lasers, lagers*
    but *trial lawers* → *trial lawyers*
- Run-on errors
  - e.g., "homestreetbank"
  - missing spaces can be considered as a single character error

Noisy Channel Model .. 1

User chooses word *w* based on probability distribution *P(w)*

called the *language model*

can capture context information, e.g. $P(w_1|w_2)$

User writes word, but noisy channel causes word *e* to be written instead with probability *P(e|w)*

called *error model*

represents information about the frequency of spelling errors

Noisy Channel Model .. 2

Need to estimate probability of correction

$P(w|e) = P(e|w)P(w)$

Estimate language model using context

e.g., $P(w) = \lambda P(w) + (1 - \lambda)P(w|w_p)$

w_p is previous word

Example:

"fish tink"

"tank" and "think" both likely corrections,
but $P(\text{tank}|\text{fish}) > P(\text{think}|\text{fish})$

Noisy Channel Model .. 3

Language model probabilities estimated using corpus and query log

Both simple and complex methods used for estimating error model

simple approach: assume all words with same edit distance have same probability, only edit distance 1 and 2 considered

complex approach: incorporate estimates of:

- common typing errors (ie vs ei, teh vs. the, etc.)
- also take into account keyboard layout (n-m, a-s, l-, etc.)

Example Spellcheck Process (Cucerzan & Brill)

- Tokenize the query
- For each token, find set of alternative words based on certain types of errors, using dictionary based on both query log and trusted dictionary.
- Noisy channel model used to select best correction.
- Continue looking for alternatives, finding best correction, till no better correction found.

e.g.

miniture golfcurses
miniature golfcourses
miniature golf courses

from Sylviu Cucerzan's Facebook page

Peter Norvig's Spell Checker .. 1

```
import re
from collections import Counter

def words(text):
       return re.findall(r'\w+', text.lower())

WORDS = Counter(words(open('big.txt').read()))

def P(word, N=sum(WORDS.values())):
    "Probability of `word`."
    return WORDS[word] / N

def correction(word):
    "Most probable spelling correction for word."
    return max(candidates(word), key=P)

def candidates(word):
    "Generate possible spelling corrections for word."
    return (known([word]) or known(edits1(word))
           or known(edits2(word)) or [word])
```

from Peter Norvig's Facebook page

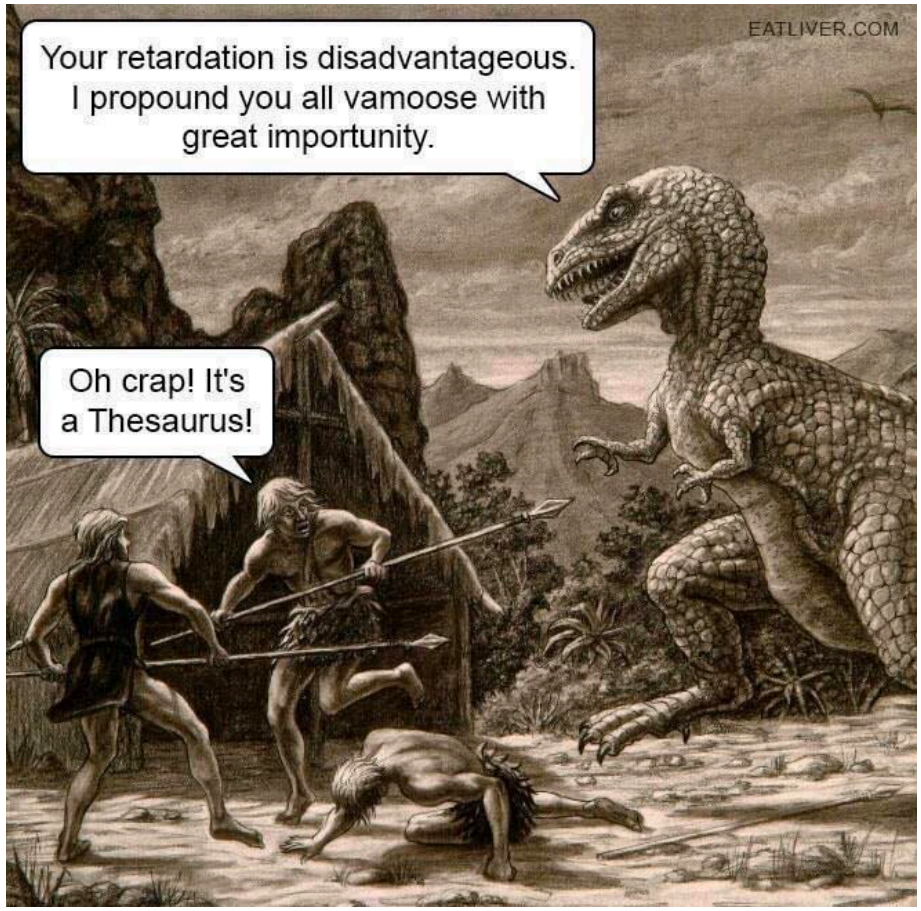
Peter Norvig's Spell Checker .. 2

```
def known(words):
    "The subset of `words` that appear in the dictionary of WORDS."
    return set(w for w in words if w in WORDS)

def edits1(word):
    "All edits that are one edit away from `word`."
    letters    = 'abcdefghijklmnopqrstuvwxyz'
    splits     = [(word[:i], word[i:])    for i in range(len(word) + 1)]
    deletes    = [L + R[1:]               for L, R in splits if R]
    transposes = [L + R[1] + R[0] + R[2:] for L, R in splits if len(R)>1]
    replaces   = [L + c + R[1:]           for L, R in splits if R for c in
                                              letters]
    inserts    = [L + c + R               for L, R in splits for c in
                                              letters]
    return set(deletes + transposes + replaces + inserts)

def edits2(word):
    "All edits that are two edits away from `word`."
    return (e2 for e1 in edits1(word) for e2 in edits1(e1))
```

QUERY EXPANSION

The Thesaurus

- Used in early search engines as a tool for *indexing* and *query formulation*
  - specifies preferred terms and relationships between them
  - also called *controlled vocabulary*
- Particularly useful for *query expansion*
  - adding synonyms or more specific terms using a thesaurus
  - improves search effectiveness



Facebook

MeSH (Medical Subject Headings) Thesaurus

MeSH Heading	Neck Pain
Tree Number	C10.597.617.576
Tree Number	C23.888.592.612.553
Tree Number	C23.888.646.501
Entry Term	Cervical Pain
Entry Term	Neckache
Entry Term	Anterior Cervical Pain
Entry Term	Anterior Neck Pain
Entry Term	Cervicalgia
Entry Term	Cervicodynia
Entry Term	Neck Ache
Entry Term	Posterior Cervical Pain
Entry Term	Posterior Neck Pain

https://www.nlm.nih.gov/mesh/

Query Expansion

A variety of *automatic* or *semi-automatic* query expansion techniques have been developed

goal: improve effectiveness by matching related terms

semi-automatic techniques require user interaction to select best expansion terms

Query suggestion is a related technique

alternative queries, not necessarily more terms

Query Expansion

Approaches usually based on an analysis of term co-occurrence

either in the entire document collection, a large collection of queries, or the top-ranked documents in a result list

query-based stemming also an expansion technique

Automatic expansion based on general thesaurus not effective

does not take context into account
e.g. bank, lead

Term Association Measures .. 1

- *Dice's Coefficient*

$$\frac{2.n_{ab}}{n_a+n_b} \overset{rank}{=} \frac{n_{ab}}{n_a+n_b}$$

- *Mutual Information*

$$\log \frac{P(a,b)}{P(a)P(b)} = \log N.\frac{n_{ab}}{n_a.n_b} \overset{rank}{=} \frac{n_{ab}}{n_a.n_b}$$

assuming N documents/windows

What if: $n_a = n_b = 10$, $n_{ab} = 5$. MI = $5/(10*10) = 5 * 10^{-2}$

But if $n_a = n_b = 1000$, $n_{ab} = 500$. MI = $500/(1000*1000) = 5 * 10^{-4}$!

Term Association Measures .. 2

- Mutual Information measure favors low frequency terms, so use

- *Expected Mutual Information Measure* (EMIM)

$$P(a,b).\log\frac{P(a,b)}{P(a)P(b)} = \frac{n_{ab}}{N}\log(N.\frac{n_{ab}}{n_a.n_b}) \overset{rank}{=} n_{ab}.\log(N.\frac{n_{ab}}{n_a.n_b})$$

  - actually only a part of full EMIM, focused on word occurrence

Term Association Measures .. 3

- *Pearson's Chi-squared (χ^2) measure*
  - compares the number of co-occurrences of two words with the expected number of co-occurrences if the two words were independent
  - normalizes this comparison by the expected number
  - also limited form focused on word co-occurrence

$$\frac{(n_{ab} - N.\frac{n_a}{N}.\frac{n_b}{N})^2}{N.\frac{n_a}{N}.\frac{n_b}{N}} \underline{=} rank \frac{(n_{ab} - \frac{1}{N}.n_a.n_b)^2}{n_a.n_b}$$

Association Measures: Summary

Measure	*Formula*
Mutual information (MIM)	$\frac{n_{ab}}{n_a.n_b}$
Expected Mutual Information $(EMIM)$	$n_{ab}.\log(N.\frac{n_{ab}}{n_a.n_b})$
Chi-square (χ^2)	$\frac{(n_{ab}-\frac{1}{N}.n_a.n_b)^2}{n_a.n_b}$
Dice's coefficient $(Dice)$	$\frac{n_{ab}}{n_a+n_b}$

Association Measure Example .. 1

MIM	*EMIM*	χ^2	*Dice*
trmm	forest	trmm	forest
itto	tree	itto	exotic
ortuno	rain	ortuno	timber
kuroshio	island	kuroshio	rain
ivirgarzama	like	ivirgarzama	banana
biofunction	fish	biofunction	deforestation
kapiolani	most	kapiolani	plantation
bstilla	water	bstilla	coconut
almagreb	fruit	almagreb	jungle
jackfruit	area	jackfruit	tree
adeo	world	adeo	rainforest
xishuangbanna	america	xishuangbanna	palm
frangipani	some	frangipani	hardwood
yuca	live	yuca	greenhouse
anthurium	plant	anthurium	logging

Most strongly associated words for “tropical” in a collection of TREC news stories. Co-occurrence counts are measured at the document level.

Association Measure Example .. 2

MIM	*EMIM*	χ^2	*Dice*
zoologico	water	arlsq	species
zapanta	species	happyman	wildlife
wrint	wildlife	outerlimit	fishery
wpfmc	fishery	sportk	water
weighout	sea	lingcod	fisherman
waterdog	fisherman	longfin	boat
longfin	boat	bontadelli	sea
veracruzana	area	sportfisher	habitat
ungutt	habitat	billfish	vessel
ulocentra	vessel	needlefish	marine
needlefish	marine	damaliscu	endanger
tunaboat	land	bontebok	conservation
tsolwana	river	taucher	river
olivacea	food	orangemouth	catch
motoroller	endanger	sheepshead	island

Most strongly associated words for “fish” in a collection of TREC news stories.

Association Measure Example .. 3

MIM	*EMIM*	χ^2	*Dice*
zapanta	wildlife	gefilte	wildlife
plar	vessel	mbmo	vessel
mbmo	boat	zapanta	boat
gefilte	fishery	plar	fishery
hapc	species	hapc	species
odfw	tuna	odfw	catch
southpoint	trout	southpoint	water
anadromous	fisherman	anadromous	sea
taiffe	salmon	taiffe	meat
mollie	catch	mollie	interior
frampton	nmf	frampton	fisherman
idfg	trawl	idfg	game
billingsgate	halibut	billingsgate	salmon
sealord	meat	sealord	tuna
longline	shellfish	longline	caught

Most strongly associated words for “fish” in a collection of TREC news stories. Co-occurrence counts are measured in windows of 5 words.

Association Measures

- Associated words are of little use for expanding the words in the query [tropical fish]
- Expansion based on whole query takes context into account
  - e.g., using Dice with phrase “tropical fish” gives the following highly associated words:

    goldfish, reptile, aquarium, coral, frog, exotic, stripe, regent, pet, wet
- Impractical for all possible queries (imagine an 8-word query); other approaches used to achieve this effect

Other Approaches

- Pseudo-relevance feedback
  - expansion terms based on top retrieved documents for initial query
- Context vectors
  - Represent words by the words that co-occur with them
    - e.g., top 35 most strongly associated words for “aquarium” (using Dice’s coefficient):

      zoology, cranmore, jouett, zoo, goldfish, fish, cannery, urchin, reptile, coral, animal, mollusk, marine, underwater, plankton, mussel, oceanography, mammal, species, exhibit, swim, biologist, cabrillo, saltwater, creature, reef, whale, oceanic, scuba, kelp, invertebrate, park, crustacean, wild, tropical

  - Rank words for a query by ranking context vectors

Other Approaches

Query logs

- Best source of information about queries and related terms
  - short pieces of text and click data
- e.g., most frequent words in queries containing “tropical fish” from a search log:

  stores, pictures, live, sale, types, clipart, blue, freshwater, aquarium, supplies
- query suggestion based on finding similar queries
  - group based on click data

Queries → Clicked Sites → Queries → Clicked Sites …

RELEVANCE FEEDBACK

Relevance Feedback

User identifies relevant (and maybe non-relevant) documents in the initial result list

System modifies query using terms from those documents and re-ranks documents

example of simple machine learning algorithm using (very little) training data

Pseudo-relevance feedback just assumes top-ranked documents are relevant – no user input

Relevance Feedback Example .. 1

1. Badmans **Tropical Fish**

   A freshwater aquarium page covering all aspects of the **tropical fish** hobby. ... to Badman's **Tropical Fish**. ... world of aquariology with Badman's **Tropical Fish**. ...

2. **Tropical Fish**

   Notes on a few species and a gallery of photos of African cichlids.

3. The **Tropical** Tank Homepage - **Tropical Fish** and Aquariums

   Info on **tropical fish** and **tropical** aquariums, large **fish** species index with ... Here you will find lots of information on **Tropical Fish** and Aquariums. ...

4. **Tropical Fish** Centre

   Offers a range of aquarium products, advice on choosing species, feeding, and health care, and a discussion board.

5. **Tropical fish** - Wikipedia, the free encyclopedia

   Tropical fish are popular aquarium **fish** , due to their often bright coloration. ... Practical Fishkeeping • **Tropical Fish** Hobbyist • Koi. Aquarium related companies: ...

6. **Tropical Fish** Find

   Home page for **Tropical Fish** Internet Directory ... stores, forums, clubs, **fish** facts, **tropical fish** compatibility and aquarium ...

7. Breeding **tropical fish**

   ... intrested in keeping and/or breeding **Tropical**, Marine, Pond and Coldwater **fish**. ... Breeding **Tropical Fish** ... breeding **tropical**, marine, coldwater & pond **fish**. ...

8. FishLore

   Includes **tropical** freshwater aquarium how-to guides, FAQs, **fish** profiles, articles, and forums.

9. Cathy's **Tropical Fish** Keeping

   Information on setting up and maintaining a successful freshwater aquarium.

10. **Tropical Fish** Place

    Tropical Fish information for your freshwater **fish** tank ... great amount of information about a great hobby, a freshwater **tropical fish** tank. ...

Top 10 documents for “tropical fish”

Relevance Feedback Example .. 2

If we assume top 10 are relevant, most frequent terms are (with frequency):

a (926), td (535), href (495), http (357), width (345), com (343), nbsp (316), www (260), tr (239), htm (233), class (225), jpg (221)

too many stopwords and HTML expressions

Use only snippets and remove stopwords

tropical (26), fish (28), aquarium (8), freshwater (5), breeding (4), information (3), species (3), tank (2), Badman's (2), page (2), hobby (2), forums (2)

Relevance Feedback Example .. 3

If document 7 ("Breeding tropical fish") is *explicitly* indicated to be relevant, the most frequent terms are:

breeding (4), fish (4), tropical (4), marine (2), pond (2), coldwater (2), keeping (1), interested (1)

Specific weights and scoring methods used for relevance feedback depend on retrieval model

Relevance Feedback

Both relevance feedback and pseudo-relevance feedback are effective, but not used in many applications

pseudo-relevance feedback has reliability issues, especially with queries that don't retrieve many relevant documents

Some applications use relevance feedback

filtering, "more like this"

Query suggestion more popular

may be less accurate, but can work if initial query fails

CONTEXT AND PERSONALIZATION

Context and Personalization

If a query has the same words as another query, results will be the same regardless of

- who submitted the query
- why the query was submitted
- where the query was submitted
- what other queries were submitted in the same session
- when the query was submitted …

These other factors (the *context*) *could* have a significant impact on relevance

- difficult to incorporate into ranking

User Models

- Generate user profiles based on documents that the person looks at
  - such as web pages visited, email messages, or word processing documents on the desktop
- Modify queries using words from profile
- Effectiveness?
  - good in specific domains
  - hard for generic search; information needs may change significantly

Query Logs

Query logs provide important contextual information that can be used effectively

Context in this case is

- previous queries that are the same
- previous queries that are similar
- query sessions including the same query

Query history for individuals could be used for caching

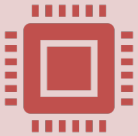
Local Search

Location is context

Local search uses geographic information to modify the ranking of search results

location derived from the query text

location of the device where the query originated



Examples:

[pizza], from a mobile phone in Redmond

Red Mill Burgers, from an Alexa device in Seattle

Paris, when you're in Texas vs. in France

Local Search

Identify the geographic region associated with web pages

use location metadata that has been manually added to the document,

or identify locations such as place names, city names, or country names in text

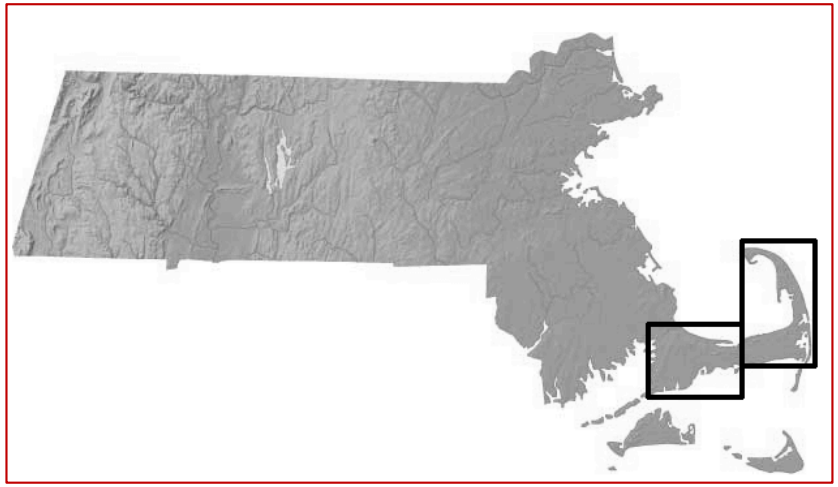
Identify the geographic region associated with the query

10-15% of queries contain some location reference

Rank web pages using location information in addition to text and link-based features

Extracting Location Information

- Type of information extraction
  - ambiguity and significance of locations are issues
- Location names are mapped to specific regions and coordinates



- Matching done by inclusion, distance

Summary

Looked at a variety of ways in which users can alter the results they get, with:

- Query-based Stemming
- Spell Checking & Spelling Suggestions
- Query Expansion
- Relevance Feedback & Pseudo-Relevance Feedback
- Context & Personalization

Next week: **Retrieval Models 1**

Readings

Croft, Metzler & Strohman (CMS),
Chapter 6, Sections 6.1, 6.2

Optional:

Peter Norvig, How to Write a Spelling Corrector,
http://norvig.com/spell-correct.html

Manning, Raghavan & Schütze (MRS), Sections 3.3, 3.4,
Chapter 9